PSTAT 131 Final Project: Model comparison for predicting the salary of a data science job

Nicholas Axl Andrian

2023-11-23

Dataset used: <https://www.kaggle.com/datasets/arnabchaki/data-science-salaries-2023>

In this project, we will be fitting several different machine learning algorithms to find out which method of prediction is the most accurate in getting the predicted salary(in usd).

About the data set’s variables (excerpt from the kaggle site)

* work\_year: The year the salary was paid.
* experience\_level: The experience level in the job during the year
* employment\_type: The type of employment for the role
* job\_title: The role worked in during the year.
* salary: The total gross salary amount paid.
* salary\_currency: The currency of the salary paid as an ISO 4217 currency code.
* salaryinusd: The salary in USD
* employee\_residence: Employee’s primary country of residence in during the work + year as an ISO 3166 country code.
* remote\_ratio: The overall amount of work done remotely
* company\_location: The country of the employer’s main office or contracting branch
* company\_size: The median number of people that worked for the company during the year

library(dplyr)  
library(randomForest)  
library(gbm)  
library(ISLR)  
library(tree)  
library(tidyverse)  
library(ggplot2)  
library(gridExtra)

## Part 1: Exploratory Data Analysis

Loading the dataset

salaries <- read.csv("ds\_salaries.csv")

Checking the structure of the dataset

head(salaries)

## work\_year experience\_level employment\_type job\_title salary  
## 1 2023 SE FT Principal Data Scientist 80000  
## 2 2023 MI CT ML Engineer 30000  
## 3 2023 MI CT ML Engineer 25500  
## 4 2023 SE FT Data Scientist 175000  
## 5 2023 SE FT Data Scientist 120000  
## 6 2023 SE FT Applied Scientist 222200  
## salary\_currency salary\_in\_usd employee\_residence remote\_ratio  
## 1 EUR 85847 ES 100  
## 2 USD 30000 US 100  
## 3 USD 25500 US 100  
## 4 USD 175000 CA 100  
## 5 USD 120000 CA 100  
## 6 USD 222200 US 0  
## company\_location company\_size  
## 1 ES L  
## 2 US S  
## 3 US S  
## 4 CA M  
## 5 CA M  
## 6 US L

## 'data.frame': 3755 obs. of 11 variables:  
## $ work\_year : int 2023 2023 2023 2023 2023 2023 2023 2023 2023 2023 ...  
## $ experience\_level : chr "SE" "MI" "MI" "SE" ...  
## $ employment\_type : chr "FT" "CT" "CT" "FT" ...  
## $ job\_title : chr "Principal Data Scientist" "ML Engineer" "ML Engineer" "Data Scientist" ...  
## $ salary : int 80000 30000 25500 175000 120000 222200 136000 219000 141000 147100 ...  
## $ salary\_currency : chr "EUR" "USD" "USD" "USD" ...  
## $ salary\_in\_usd : int 85847 30000 25500 175000 120000 222200 136000 219000 141000 147100 ...  
## $ employee\_residence: chr "ES" "US" "US" "CA" ...  
## $ remote\_ratio : int 100 100 100 100 100 0 0 0 0 0 ...  
## $ company\_location : chr "ES" "US" "US" "CA" ...  
## $ company\_size : chr "L" "S" "S" "M" ...

Already we can see an issue that needs to be worked on. Several variables seem to supposedly be read in as factors. We will finish conducting checks on the dataset before converting said columns.

Checking the summary of the dataset

## work\_year experience\_level employment\_type job\_title   
## Min. :2020 Length:3755 Length:3755 Length:3755   
## 1st Qu.:2022 Class :character Class :character Class :character   
## Median :2022 Mode :character Mode :character Mode :character   
## Mean :2022   
## 3rd Qu.:2023   
## Max. :2023   
## salary salary\_currency salary\_in\_usd employee\_residence  
## Min. : 6000 Length:3755 Min. : 5132 Length:3755   
## 1st Qu.: 100000 Class :character 1st Qu.: 95000 Class :character   
## Median : 138000 Mode :character Median :135000 Mode :character   
## Mean : 190696 Mean :137570   
## 3rd Qu.: 180000 3rd Qu.:175000   
## Max. :30400000 Max. :450000   
## remote\_ratio company\_location company\_size   
## Min. : 0.00 Length:3755 Length:3755   
## 1st Qu.: 0.00 Class :character Class :character   
## Median : 0.00 Mode :character Mode :character   
## Mean : 46.27   
## 3rd Qu.:100.00   
## Max. :100.00

Checking for null values

colSums(is.na(salaries))

## work\_year experience\_level employment\_type job\_title   
## 0 0 0 0   
## salary salary\_currency salary\_in\_usd employee\_residence   
## 0 0 0 0   
## remote\_ratio company\_location company\_size   
## 0 0 0

Fortunately, we have no null values so imputing is not required

Checking potential factor columns for their unique values

factor\_cols <- salaries[, c(2, 3, 4, 6, 8, 10, 11)]  
  
# finding unique values, referenced code from https://www.kaggle.com/code/abdulfaheem11/data-science-salaries-2023-analysis  
  
# output ommitted to prevent too much space being taken up  
sapply(factor\_cols, function(col) unique(col))

Changing said variables to become factors

salaries[, c(2, 3, 4, 6, 8, 10, 11)] <- lapply(factor\_cols, factor)  
str(salaries)

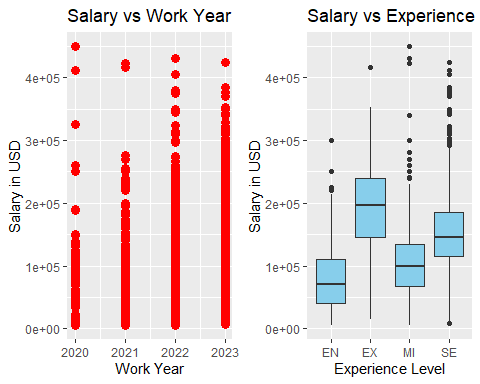
We can also drop the salary and salary\_currency as we will just be using the salary\_in\_usd to simplify our steps. We will also drop the employee\_residence, to put more focus onto the company\_location instead.

salaries <- salaries[, !(names(salaries) %in% c('salary\_currency','salary', 'employee\_residence'))]

Visualization to search for patterns with regards to the salary\_in\_usd

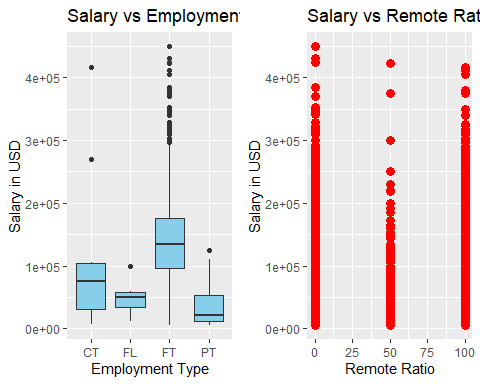
Prioritizing focus on work\_year, experience\_level, employment\_type, job\_title, employee\_residence, remote\_ratio, company\_location, company\_size

yearplot <- ggplot(salaries, aes(x = work\_year, y = salary\_in\_usd)) +  
 geom\_point(color = "red", size = 3) +  
 labs(x = "Work Year", y = "Salary in USD", title = "Salary vs Work Year")  
expplot <- ggplot(salaries, aes(x = experience\_level, y = salary\_in\_usd)) +  
 geom\_boxplot(fill = "skyblue") +  
 labs(x = "Experience Level", y = "Salary in USD", title = "Salary vs Experience Level")  
grid.arrange(yearplot, expplot, ncol = 2)



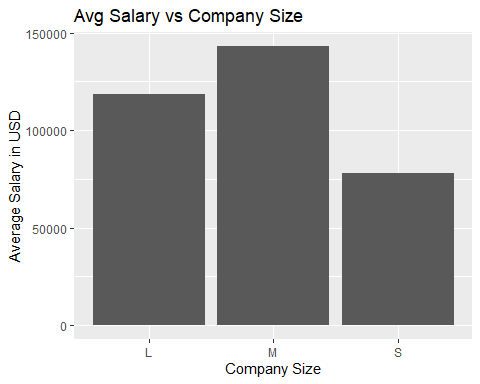
* We can see that the average salary in usd increases as the years go by, as the line congests further upwards towards the end.
* Experience level does not really show much of a trend as it goes towards seniority, We can tell though that EX has the highest average and MI has the highest peak

employplot <- ggplot(salaries, aes(x = employment\_type, y = salary\_in\_usd)) +  
 geom\_boxplot(fill = "skyblue") +  
 labs(x = "Employment Type", y = "Salary in USD", title = "Salary vs Employment Type")  
remoteplot <- ggplot(salaries, aes(x = remote\_ratio, y = salary\_in\_usd)) +  
 geom\_point(color = "red", size = 3, shape = 19) +  
 labs(x = "Remote Ratio", y = "Salary in USD", title = "Salary vs Remote Ratio")  
grid.arrange(employplot, remoteplot, ncol = 2)



* In employment type, FT has the highest average as well as higher peaks
* Remote ratio consists of 0, 50 and 100. The highest points as well as average are in the order 0>100>50

usd\_salary\_by\_size <- salaries%>%  
 group\_by(company\_size)%>%  
 summarise(Avg\_sal=mean(salary\_in\_usd))  
  
sizeplot <- ggplot(usd\_salary\_by\_size, aes(x=company\_size, y=Avg\_sal)) +  
 geom\_col() +  
 labs(title='Avg Salary vs Company Size', x='Company Size', y='Average Salary in USD')  
sizeplot

 + From this plot we can also see that medium sized companies pay the largest on average, followed by large then small

Tabling the 6 highest and lowest paying jobs

options("max.print" = 6)  
top\_6\_job\_salaries<-salaries%>%  
 group\_by(job\_title)%>%  
 summarise(Avg\_Sal=mean(salary\_in\_usd))%>%  
 arrange(desc(Avg\_Sal))%>%  
 head()  
top\_6\_job\_salaries

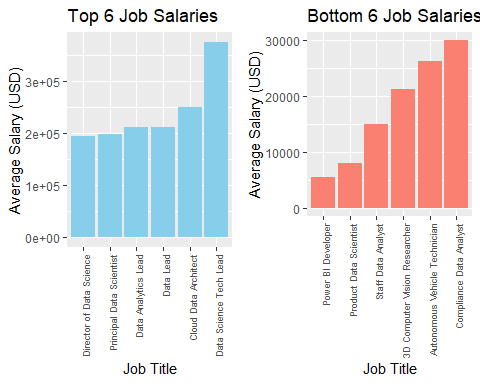
## # A tibble: 6 × 2  
## job\_title Avg\_Sal  
## <fct> <dbl>  
## 1 Data Science Tech Lead 375000   
## 2 Cloud Data Architect 250000   
## 3 Data Lead 212500   
## 4 Data Analytics Lead 211254.  
## 5 Principal Data Scientist 198171.  
## 6 Director of Data Science 195141.

bottom\_6\_job\_salaries<-salaries%>%  
 group\_by(job\_title)%>%  
 summarise(Avg\_Sal=mean(salary\_in\_usd))%>%  
 arrange(Avg\_Sal)%>%  
 head()  
bottom\_6\_job\_salaries

## # A tibble: 6 × 2  
## job\_title Avg\_Sal  
## <fct> <dbl>  
## 1 Power BI Developer 5409   
## 2 Product Data Scientist 8000   
## 3 Staff Data Analyst 15000   
## 4 3D Computer Vision Researcher 21352.  
## 5 Autonomous Vehicle Technician 26278.  
## 6 Compliance Data Analyst 30000

Plotting the 6 highest and lowest paying jobs

top6jobplot <- ggplot(top\_6\_job\_salaries, aes(x = reorder(job\_title, Avg\_Sal), y = Avg\_Sal)) +  
 geom\_bar(stat = "identity", fill = "skyblue") +  
 labs(title = "Top 6 Job Salaries", x = "Job Title", y = "Average Salary (USD)")  
bot6jobplot <- ggplot(bottom\_6\_job\_salaries, aes(x = reorder(job\_title, Avg\_Sal), y = Avg\_Sal)) +  
 geom\_bar(stat = "identity", fill = "salmon") +  
 labs(title = "Bottom 6 Job Salaries", x = "Job Title", y = "Average Salary (USD)")  
top6jobplot <- top6jobplot + theme(axis.text.x = element\_text(angle = 90, vjust = 0.75, size=7, hjust=1))  
bot6jobplot <- bot6jobplot + theme(axis.text.x = element\_text(angle = 90, vjust = 0.75, size=7, hjust=1))  
grid.arrange(top6jobplot,bot6jobplot, ncol = 2)

 + We can tell that the data science tech lead job has the highest average pay by far + we can also tell that the power bi developer has the lowest pay out of all the jobs

Tabling the 6 highest and lowest paying company locations

top\_6\_loc<-salaries%>%  
 group\_by(company\_location)%>%  
 summarise(Avg\_Sal=mean(salary\_in\_usd))%>%  
 arrange(desc(Avg\_Sal))%>%  
 head()  
top\_6\_loc

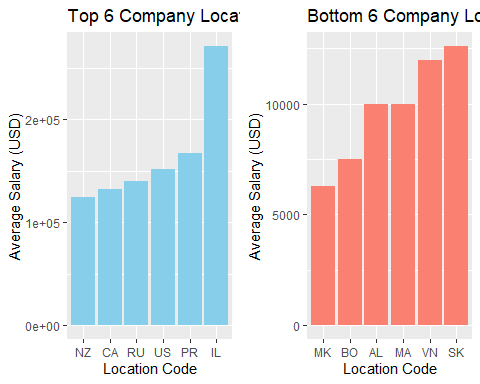
## # A tibble: 6 × 2  
## company\_location Avg\_Sal  
## <fct> <dbl>  
## 1 IL 271446.  
## 2 PR 167500   
## 3 US 151822.  
## 4 RU 140333.  
## 5 CA 131918.  
## 6 NZ 125000

bottom\_6\_loc<-salaries%>%  
 group\_by(company\_location)%>%  
 summarise(Avg\_Sal=mean(salary\_in\_usd))%>%  
 arrange(Avg\_Sal)%>%  
 head()  
bottom\_6\_loc

## # A tibble: 6 × 2  
## company\_location Avg\_Sal  
## <fct> <dbl>  
## 1 MK 6304  
## 2 BO 7500  
## 3 AL 10000  
## 4 MA 10000  
## 5 VN 12000  
## 6 SK 12608

Plotting the 6 highest and lowest paying company locations

top6locplot <- ggplot(top\_6\_loc, aes(x = reorder(company\_location, Avg\_Sal), y = Avg\_Sal)) +  
 geom\_bar(stat = "identity", fill = "skyblue") +  
 labs(title = "Top 6 Company Locations", x = "Location Code", y = "Average Salary (USD)")  
bot6locplot <- ggplot(bottom\_6\_loc, aes(x = reorder(company\_location, Avg\_Sal), y = Avg\_Sal)) +  
 geom\_bar(stat = "identity", fill = "salmon") +  
 labs(title = "Bottom 6 Company Locations", x = "Location Code", y = "Average Salary (USD)")  
top6resplot <- top6locplot + theme(axis.text.x = element\_text(angle = 90, vjust = 0.75, size=7, hjust=1))  
bot6resplot <- bot6locplot + theme(axis.text.x = element\_text(angle = 90, vjust = 0.75, size=7, hjust=1))  
grid.arrange(top6locplot,bot6locplot, ncol = 2)

 + It seems that IL has the highest paying jobs when it comes to company location + MK has the lowest paying job out of all the locations by far

## Part 2: Problem formulation and discussion of statistical learning algorithms used

The main goal of this project, as mentioned before is to compare several statistical learning methods in predicting the salary given the parameters. We will be focusing on the following methods:

* K-Nearest Neighbors
  + Using K-NN means
  + Scaling the predictors
* Linear Regression
  + Calculating the accuracy of the Least Squares Estimate
* Ridge and LASSO Regression
  + Using CV methods to find the best lambda
  + Conducting variable selection with LASSO
* Regression Trees
  + Further Optimization through boosting methods
* Support Vector Machines
  + Testing SVMs with different OSH methods

Prepare